In [ ]:

```
# Assignment-7: String in Python

# 🔹 String Basics

1. Filter vowels and consonants from the string "How are you sir".

2. Count vowels and consonants in the string "How are you sir".

3. Reverse the string "How are you sir".

4. Convert lowercase letters to uppercase in the string "How are you sir".

5. Remove duplicate letters from the string "this is python programming
place".

6. Search for a specific character in the string "this is python
programming place".

7. Find the greatest and smallest characters from the string "venugopaliyer".

8. Count the total occurrences of a specific letter in the string "this is
python programming place".

9. Replace "python" with "javascript" in the string "python developer
python engineer python holder".

10.Print alternate letters from the string "How are you sir".

11. Convert the string "qwertyuiopasdfghjklzxcvbnm" to
"abcdefghijklmnopqrstuvwxyz".
```

In [1]:

```
#1. Filter vowels and consonants from the string "How are you sir".

text = "How are you sir"
vowels = "aeiouAEIOU"
vowel_list = []
consonant_list = []

for ch in text:
    if ch.isalpha():
        if ch in vowels:
            vowel_list.append(ch)
        else:
            consonant_list.append(ch)

print("Vowels:", vowel_list)
print("Consonants:", consonant_list)
```

```
Vowels: ['o', 'a', 'e', 'o', 'u', 'i']
Consonants: ['H', 'w', 'r', 'y', 's', 'r']
```

In [2]:

```
#2. Count vowels and consonants in the string "How are you sir"
text = "How are you sir"
vowels = "aeiouAEIOU"
```

```
vowel_count = 0
consonant_count = 0

for ch in text:
    if ch.isalpha():
        if ch in vowels:
            vowel_count += 1
        else:
            consonant_count += 1

print("Vowels:", vowel_count)
print("Consonants:", consonant_count)
```

```
Vowels: 6
Consonants: 6
```

In [3]:

```
#3. Reverse the string "How are you sir".

text = "How are you sir"

# slicing method
reversed_text = text[::-1]
print("Original String:", text)
print("Reversed String:", reversed_text)
```

```
Original String: How are you sir
Reversed String: ris uoy era woH
```

In [4]:

```
#4. Convert lowercase letters to uppercase in the string "How are you sir".
text = "How are you sir"

upper_text = text.upper()

print("Original String:", text)
print("Uppercase String:", upper_text)
```

```
Original String: How are you sir
Uppercase String: HOW ARE YOU SIR
```

In [5]:

```
#5. Remove duplicate letters from the string "this is python programming
#place".
text = "this is python programming place"

result = ""
for ch in text:
    if ch not in result:
        result += ch

print("Original String:", text)
print("Without Duplicates:", result)
```

```
Original String: this is python programming place
Without Duplicates: this pyonrgamlce
```

In [6]:

```
#6. Search for a specific character in the string "this is python
#programming place".
text = "this is python programming place"
ch = 'p'
```

```
if ch in text:
    print(f"'{ch}' is present in the string.")
else:
    print(f"'{ch}' is not present in the string.")
```

```
'p' is present in the string.
```

In [7]:

```
#7. Find the greatest and smallest characters from the string "venugopaliyer".
text = "venugopaliyer"

smallest = min(text)
greatest = max(text)

print("Original String:", text)
print("Smallest Character:", smallest)
print("Greatest Character:", greatest)
```

```
Original String: venugopaliyer
Smallest Character: a
Greatest Character: y
```

In [8]:

```
#8. Count the total occurrences of a specific letter in the string "this is
#python programming place"
text = "this is python programming place"
ch = 'i'

count = text.count(ch)

print(f"'{ch}' occurs {count} times in the string.")
```

```
'i' occurs 3 times in the string.
```

In [9]:

```
#9. Replace "python" with "javascript" in the string "python developer
#python engineer python holder".
text = "python developer python engineer python holder"

new_text = text.replace("python", "javascript")

print("Original String:", text)
print("Modified String:", new_text)
```

```
Original String: python developer python engineer python holder
Modified String: javascript developer javascript engineer javascript holder
```

In [10]:

```
#10.Print alternate letters from the string "How are you sir"
text = "How are you sir"
alternate = text[::2]

print("Original String:", text)
print("Alternate Letters:", alternate)
```

```
Original String: How are you sir
Alternate Letters: Hwaeyusr
```

In [11]:

```
#11. Convert the string "qwertyuiopasdfghjklzxcvbnm" to
#"abcdefghijklmnopqrstuvwxyz".
text = "qwertyuiopasdfghjklzxcvbnm"
```

```
alphabetical = "".join(sorted(text))

print("Original String:", text)
print("Alphabetical Order:", alphabetical)
```

```
Original String: qwertyuiopasdfghjklzxcvbnm
Alphabetical Order: abcdefghijklmnopqrstuvwxyz
```

In [ ]:

```
# Extra String Questions
12.Check if the string is a palindrome (e.g., "madam" → Palindrome, "hello"
→ Not palindrome).

13.Count spaces, digits, alphabets, and special characters in "Python 3.9
is awesome!!".

14.Find the longest word in the string "Python programming is
interesting".

15.Capitalize the first letter of each word in "welcome to python world".

16.Remove all spaces from "How are you sir".

17.Check if all characters in the string are unique (e.g., "abcde" → True,
"hello" → False).

18.Sort characters alphabetically in "programming" → "aggimmnoprr".

19.Swap cases of all letters in "Python Is Fun" → "pYTHON iS fUN".

20.Find frequency of each character in "banana" → { 'b':1, 'a':3,
'n':2 }.

21.Remove vowels from "How are you sir" → "Hw r y sr".

22.Check if a substring exists in "Python programming" (e.g., "thon" →
Found).

23.Print words in reverse order in "How are you sir" → "sir you are
How".

24.Count words in the string "This is a python assignment".

25.Find the ASCII value of each character in "ABcd".

26.Convert a string into a list of words using "split()" (e.g., "Python is
fun" → ["Python", "is", "fun"]).

27.Join a list of words into a string using "join()" (e.g., ["Python", "is",
"fun"] → "Python is fun").

28.Find the first non-repeating character in "swiss" → "w".

29.Check if two strings are anagrams (e.g., "listen" and "silent" →
Anagrams).
```

```
30.Replace all spaces with hyphens (-) in "Python is easy to learn" →
"Python-is-easy-to-learn".
```

In [12]:

```
#12.Check if the string is a palindrome (e.g., "madam" → Palindrome, "hello"
#→ Not palindrome).
def is_palindrome(text):
    cleaned = text.replace(" ", "").lower()
    return cleaned == cleaned[::-1]

# Test cases
print("madam →", "Palindrome" if is_palindrome("madam") else "Not Palindrome")
print("hello →", "Palindrome" if is_palindrome("hello") else "Not Palindrome")
print("nurses run →", "Palindrome" if is_palindrome("nurses run") else "Not Palindr
```

```
madam → Palindrome
hello → Not Palindrome
nurses run → Palindrome
```

In [13]:

```
#13.Count spaces, digits, alphabets, and special characters in "Python 3.9
#is awesome!!".
text = "Python 3.9 is awesome!!"

spaces = digits = alphabets = specials = 0

for ch in text:
    if ch.isspace():
        spaces += 1
    elif ch.isdigit():
        digits += 1
    elif ch.isalpha():
        alphabets += 1
    else:
        specials += 1

print("Spaces:", spaces)
print("Digits:", digits)
print("Alphabets:", alphabets)
print("Special Characters:", specials)
```

```
Spaces: 3
Digits: 2
Alphabets: 15
Special Characters: 3
```

In [14]:

```
#14.Find the longest word in the string "Python programming is
#interesting".
text = "Python programming is interesting"
words = text.split()
longest_word = max(words, key=len)

print("Original String:", text)
print("Longest Word:", longest_word)
print("Length:", len(longest_word))
```

```
Original String: Python programming is interesting
Longest Word: programming
Length: 11
```

In [15]:

```
#15.Capitalize the first letter of each word in "welcome to python world".
text = "welcome to python world"

capitalized_text = text.title()

print("Original String:", text)
print("Capitalized String:", capitalized_text)
```

```
Original String: welcome to python world
Capitalized String: Welcome To Python World
```

In [16]:

```
#16.Remove all spaces from "How are you sir".
text = "How are you sir"
no_spaces = text.replace(" ", "")
print("Original String:", text)
print("Without Spaces:", no_spaces)
```

```
Original String: How are you sir
Without Spaces: Howareyousir
```

In [17]:

```
#17.Check if all characters in the string are unique (e.g., "abcde" → True,
#"hello" → False).
def all_unique(text):
    return len(set(text)) == len(text)

# Test cases
print("abcde →", all_unique("abcde"))   # True
print("hello →", all_unique("hello"))   # False
```

```
abcde → True
hello → False
```

In [18]:

```
#18.Sort characters alphabetically in "programming" → "aggimmnoprr".
text = "programming"

sorted_text = "".join(sorted(text))

print("Original String:", text)
print("Sorted String:", sorted_text)
```

```
Original String: programming
Sorted String: aggimmnoprr
```

In [19]:

```
#19.Swap cases of all letters in "Python Is Fun" → "pYTHON iS fUN".
text = "Python Is Fun"

swapped = text.swapcase()

print("Original String:", text)
print("Swapped String:", swapped)
```

```
Original String: Python Is Fun
Swapped String: pYTHON iS fUN
```

In [20]:

```
#20.Find frequency of each character in "banana" → { 'b':1, 'a':3,
#'n':2 }.
text = "banana"
```

```
freq = {}
for ch in text:
    freq[ch] = freq.get(ch, 0) + 1

print(freq)
```

```
{'b': 1, 'a': 3, 'n': 2}
```

In [22]:

```
#21.Remove vowels from "How are you sir" → "Hw r y sr".
text = "How are you sir"
vowels = "aeiouAEIOU"

no_vowels = "".join([ch for ch in text if ch not in vowels])

print("Original String:", text)
print("Without Vowels:", no_vowels)
```

```
Original String: How are you sir
Without Vowels: Hw r y sr
```

In [23]:

```
#22.Check if a substring exists in "Python programming" (e.g., "thon" →
#Found).
text = "Python programming"
sub = "thon"

if sub in text:
    print(f"'{sub}' → Found")
else:
    print(f"'{sub}' → Not Found")
```

```
'thon' → Found
```

In [24]:

```
#23.Print words in reverse order in "How are you sir" → "sir you are
#How".
text = "How are you sir"
reversed_words = " ".join(text.split()[::-1])

print("Original String:", text)
print("Reversed Words:", reversed_words)
```

```
Original String: How are you sir
Reversed Words: sir you are How
```

In [25]:

```
#24.Count words in the string "This is a python assignment".
text = "This is a python assignment"
word_count = len(text.split())
print("Original String:", text)
print("Word Count:", word_count)
```

```
Original String: This is a python assignment
Word Count: 5
```

In [26]:

```
#25.Find the ASCII value of each character in "ABcd"
text = "ABcd"

for ch in text:
    print(f"{ch} → {ord(ch)}")
```

```
A → 65
B → 66
c → 99
d → 100
```

In [27]:

```python
#27.Join a list of words into a string using "join()" (e.g., ["Python", "is",
#"fun"] → "Python is fun").
words = ["Python", "is", "fun"]

sentence = " ".join(words)

print("Word List:", words)
print("Joined String:", sentence)
```

```
Word List: ['Python', 'is', 'fun']
Joined String: Python is fun
```

In [28]:

```python
#28.Find the first non-repeating character in "swiss" → "w".
text = "swiss"

for ch in text:
    if text.count(ch) == 1:
        print("First non-repeating character:", ch)
        break
else:
    print("No non-repeating character found")
```

```
First non-repeating character: w
```

In [29]:

```python
#29.Check if two strings are anagrams (e.g., "listen" and "silent" →
#Anagrams).

def are_anagrams(str1, str2):
    return sorted(str1) == sorted(str2)
print("listen & silent →", "Anagrams" if are_anagrams("listen", "silent") else "Not
print("hello & world →", "Anagrams" if are_anagrams("hello", "world") else "Not Ana
```

```
listen & silent → Anagrams
hello & world → Not Anagrams
```

In [30]:

```python
#30.Replace all spaces with hyphens (-) in "Python is easy to learn" →
#"Python-is-easy-to-learn".

text = "Python is easy to learn"

new_text = text.replace(" ", "-")

print("Original String:", text)
print("Modified String:", new_text)
```

```
Original String: Python is easy to learn
Modified String: Python-is-easy-to-learn
```

In [ ]:

```python
# Substring Related Questions

31.Extract a substring from "Python Programming" → from index 0 to 6 should
give "Python".
```

```
32.Check if one string is a substring of another (e.g., "gram" is a substring of
"Programming").

33.Find all occurrences of a substring in "This is Python and Python is
fun" → Substring "Python".

34.Replace a substring in "I like Python" → Replace "Python" with
"Java".

35.Remove a substring from "HelloWorld" → Remove "World" → "Hello".

36.Count occurrences of a substring in "banana" → Substring "ana" appears 2
times.
```

In [31]:

```
#31.Extract a substring from "Python Programming" → from index 0 to 6 should
#give "Python".
text = "Python Programming"

substring = text[0:6]

print("Original String:", text)
print("Substring (0-6):", substring)
```

```
Original String: Python Programming
Substring (0-6): Python
```

In [32]:

```
#32.Check if one string is a substring of another (e.g., "gram" is a substring of
#"Programming").
text = "Programming"
sub = "gram"

if sub in text:
    print(f'"{sub}" is a substring of "{text}"')
else:
    print(f'"{sub}" is NOT a substring of "{text}"')
```

```
"gram" is a substring of "Programming"
```

In [33]:

```
#33.Find all occurrences of a substring in "This is Python and Python is
#fun" → Substring "Python".
text = "This is Python and Python is fun"
sub = "Python"

positions = []
start = 0

while True:
    index = text.find(sub, start)
    if index == -1:
        break
    positions.append(index)
    start = index + 1

print(f'Substring "{sub}" found at positions:', positions)
```

```
Substring "Python" found at positions: [8, 19]
```

In [34]:

```
#34.Replace a substring in "I like Python" → Replace "Python" with
#"Java"
text = "I like Python"

new_text = text.replace("Python", "Java")

print("Original String:", text)
print("Modified String:", new_text)
```

```
Original String: I like Python
Modified String: I like Java
```

In [35]:

```
#35.Remove a substring from "HelloWorld" → Remove "World" → "Hello".
text = "HelloWorld"

new_text = text.replace("World", "")

print("Original String:", text)
print("Modified String:", new_text)
```

```
Original String: HelloWorld
Modified String: Hello
```

In [36]:

```
#36.Count occurrences of a substring in "banana" → Substring "ana" appears 2
#times.
text = "banana"
sub = "ana"

count = text.count(sub)
print(f'Substring "{sub}" appears {count} times.')
```

```
Substring "ana" appears 1 times.
```